On Continuation Methods for the Numerical Treatment of Multi-Objective Optimization Problems

Oliver Schütze, Alessandro Dell'Aere and Michael Dellnitz

University of Paderborn,
Faculty for Computer Science, Electrical Engineering and Mathematics
Institute for Mathematics
33098 Paderborn, Warburger Strasse 100, Germany
{schuetze,dellaere,dellnitz}@uni-paderborn.de

Abstract. In this report we describe how continuation methods can be used for the numerical treatment of multi-objective optimization problems (MOPs): starting with a given Karush-Kuhn-Tucker point (KKT-point) $\tilde{x}$ of an MOP, these techniques can be applied to detect further KKT-points in the neighborhood of $\tilde{x}$. In the next step, again further points are computed starting with these new-found KKT-points, and so on. In order to maintain a good spread of these solutions we use *boxes* for the representation of the computed parts of the solution set. Based on this background, we propose a new predictor-corrector variant and show some numerical results indicating the strength of the method, in particular in higher dimensions. Further, the data structure allows for an efficient computation of solution sets of MOPs with more than two objectives, which has not been considered so far in most other existing continuation methods.

1 Introduction

In a variety of applications in industry and finance the problem arises that several objective functions have to be optimized concurrently. For instance, for a perfect economical production plan, the ultimate desire would be to simultaneously *minimize cost* and *maximize quality*. This example already illustrates a natural feature of these problems, namely that the different objectives typically contradict each other and therefore certainly do not have identical optima. Thus, the question arises how to approximate a particular "optimal compromise" (see e.g. [20] for an overview of widely used *interactive methods*) or how to compute *all* optimal compromises of this *multi–objective optimization problem* (MOP). The latter will be the topic of this article.

Mathematically speaking, in an MOP there are given k objective functions $f_1, \ldots, f_k : \mathbb{R}^n \to \mathbb{R}$ which have to be minimized. The set of optimal compromises with respect to the objective functions is called the *Pareto set*[1]. A point $x \in \mathbb{R}^n$ in parameter space is said to be a *Pareto point* if there is no other point which is at least as good as x in all the objectives and strictly better in at least one objective. Thus, in this article we will concentrate on the approximation of the Pareto set.

Multi–objective optimization is currently a very active area of research. By far most of the methods for the computation of single Pareto points or the entire Pareto set are based on a "scalarization" of the MOP (see e.g. [28] [33] [4] and [12]). For a survey of these and further methods we refer to [20] for nonlinear MOPs and to [17] and [32] in the linear case. Another way to attack the problem is by using *bio-inspired heuristics* like *Evolutionary Algorithms* (see [36] [6] [7] [5] [35] [13]) or *Particle Swarm Optimization* (see [3] [11] [21]). These methods are particularly advantageous in the situation where the MOP is discrete.

A method which is based on a stochastic approach is presented in [27]. In this work the authors derive a stochastic differential equation (SDE) which has the property that it is very likely to observe corresponding solutions in a neighborhood of the set of KKT-points. Similar to the evolutionary strategies here the idea is to directly approximate the entire solution set and not just single Pareto points on the set.

Another way to compute the entire Pareto set is to use *subdivision techniques* (see [31] [8] [30]). These algorithms start with a compact subset $Q \subset \mathbb{R}^n$ of the domain and generate outer approximations of the Pareto set which get finer under iteration (see [9] for a convergence result). The approach is of global nature and hence in practice restricted to moderate dimensions of the parameter space. Typically – that is under mild regularity conditions – the set of Pareto points is locally a $(k-1)$-dimensional manifold if there are k smooth objective functions. The aim of this report is to demonstrate that *continuation methods* can be used to compute these solution sets efficiently. However, it has to be mentioned that these techniques are of local nature: given an initial set $\mathcal{S}_0$ of KKT-points, all further solutions computed by these methods are restricted to the connected components of the set of KKT-points which contain a point $s \in \mathcal{S}_0$. Continuation methods have been thoroughly analyzed over the last three decades, see e.g. [26], [1] and [15] for the computation of general implicitly defined manifolds. In the context of multi-objective optimization these techniques have for instance been used in [14], [24], [16] and [30].

In this report we follow the ideas of [30] and present a predictor-corrector algorithm which in particular allows for the efficient computation of KKT-points for more than two objectives and demonstrate the efficiency of this approach by several examples.

An outline of this report is as follows: in Section 2 we give the required background for the algorithm for the computation of the set of KKT-points

---

[1] Named after the economist Vilfredo Pareto, 1848-1923.

which is presented in Section 3. In Section 4 we present some numerical results and finally make a conclusion in Section 5.

2   Theoretical Background

In this section we give the required background for the predictor-corrector algorithm which is described in the next chapter: we address the basic idea of continuaton methods (following mainly [26] and [1]) and give the connection to multi-objective optimization.

Continuation Methods

Assume we are given a differentiable mapping

$$H : S \subset \mathbb{R}^{N+K} \to \mathbb{R}^{N}, \quad K \geq 1, \tag{1}$$

of class $C^r, r > 1$, on an open subset $S \subset \mathbb{R}^{N+K}$. A point $x \in S$ is called *regular* if the first derivative, $H'(x)$, has full rank N. We are interested in the following system of (nonlinear) equations:

$$H(x) = 0, \quad x \in S. \tag{2}$$

In the case that the regular solution set

$$\mathcal{M} = \{x \in S \mid H(x) = 0, x \text{ regular}\} \tag{3}$$

is non-empty, it is well-known that $\mathcal{M}$ is a K-dimensional C^r-manifold in R^{N+K} without boundary (see e.g., [26]).

   A common way to solve (2) numerically is to use *continuation methods*. These methods address the following (local and discretized) problem: given an initial (approximated) solution $x^* \in \mathcal{M}$ find further solutions $x_i^* \in \mathcal{M}$ near x^*. For many applications, this problem can be solved by using a *predictor-corrector* procedure. These procedures consist of two steps which can in general be described as follows:

(P) Predict a set $\{p_1, \ldots, p_s\}$ of distinct (and well distributed) points which are near both to x^* and to $\mathcal{M}$.

(C) For $i = 1, \ldots, s$
       Starting with the predicted point p_i, compute by some (typically few) iterative steps an approximated element x_i^* of $\mathcal{M}$, i.e. $H(x_i^*) \approx 0$.

   In order to obtain well distributed predictors near to a solution $x \in \mathcal{M}$ one can compute an orthonormal basis of the tangent space at x via a QR-decomposition of $H'(x)^T$:
The tangent space at a point $x \in \mathcal{M}$ is given by

$$T_x\mathcal{M} = \ker H'(x) = \{u \in \mathbb{R}^{N+K} \mid H'(x)u = 0\}. \tag{4}$$

The normal space $N_x\mathcal{M}$ at $x \in \mathcal{M}$ is the orthogonal complement of $T_x\mathcal{M}$:

$$N_x\mathcal{M} = (T_x\mathcal{M})^\perp = (\ker H'(x))^\perp = \text{rge}\, H'(x)^T. \tag{5}$$

Let $Q = (Q_N \,|\, Q_K) \in \mathbb{R}^{(N+K)\times(N+K)}$ be an orthogonal matrix and $R = \left(\dfrac{R_N}{0}\right) \in \mathbb{R}^{(N+K)\times N}$, where $R_N \in \mathbb{R}^{N\times N}$ is an upper triangular matrix, such that

$$H'(x)^T = QR = (Q_N \,|\, Q_K)\left(\frac{R_N}{0}\right). \tag{6}$$

If x is regular it follows that the diagonal elements of R_N do not vanish and hence it is straightforward to see that the columns of $Q_N \in \mathbb{R}^{(N+K)\times N}$ provide an orthonormal basis of $\text{rge}\, H'(x)^T = N_x\mathcal{M}$. Thus, an orthonormal basis of $T_x\mathcal{M}$ is given by the columns of $Q_K \in \mathbb{R}^{(N+K)\times K}$.

For the realization of the corrector step (C), typically the Gauss-Newton method

$$x_{i+1} = x_i - H'(x_i)^+ H(x_i), \quad i = 0, 1, \ldots, \tag{7}$$

where $H'(x_i)^+ \in \mathbb{R}^{(N+K)\times N}$ is the Moore-Penrose inverse of $H'(x_i)$, is applied. It is well-known that this method converges quadratically to a point $x^* \in \mathcal{M}$ if the starting vector $x_0 \in \mathbb{R}^{N+K}$ is chosen close enough to $\mathcal{M}$. We refer e.g. to [10] for a local convergence result.
If the computation and the decomposition of the Jacobian matrix $H'(x_i)$ is too costly, one can also use the so-called *chord-method*

$$x_{i+1} = x_i - H'(x_0)^+ H(x_i), \quad i = 0, 1, \ldots, \tag{8}$$

yielding linear convergence [10].

Representing the Solution Set

A problem using predictor-corrector methods – in particular for $K > 1$ – is to obtain a "global picture" of the connected component $\mathcal{M}_{B(x^*)}$ of $\mathcal{M}$ which contains the initial solution x^* and which lies within some bounded region B. That is, the task is to compute a (finite) set $\mathcal{S}_B$ of solutions, where the elements of which are roughly uniformly distributed on $\mathcal{M}_{B(x^*)}$. To achieve this, it is crucial to provide an efficient data structure for the representation of the part of $\mathcal{M}_{B(x^*)}$ that is covered (in some suitable sense) by the set of solutions already computed.
In order to solve this problem, simplicial decompositions of $\mathbb{R}^K$ are used e.g. in [2] as well as in [25]. These techniques run into problems when the dimensions N or K get large (see [15] or [1]). In [15], implicitly defined manifolds are represented as a set of overlapping spherical balls which is well suited to problems with large embedding dimension $N + K$.
Here, we propose an alternative data structure: we will use *boxes* as a tool to

maintain a spread of the solutions.

Let us assume that every parameter is restricted to a certain range, i.e.

$$a_i \leq x_i \leq b_i, \quad i = 1, \ldots, M, \tag{9}$$

where $M = N + K$. The search space thus is given by

$$Q = [a_1, b_1] \times \ldots \times [a_M, b_M] \subset \mathbb{R}^M. \tag{10}$$

Every box $B \subset \mathbb{R}^M$ can be represented by a center $c \in \mathbb{R}^M$ and a radius $r \in \mathbb{R}_+^M$ such that

$$B = B(c, r) = \{x \in \mathbb{R}^M : |x_i - c_i| \leq r_i \, \forall i = 1, \ldots, M\}.$$

The box B can be subdivided with respect to the the j-th coordinate. This division leads to two boxes $B_-(c^-, \hat{r})$ and $B_+(c^+, \hat{r})$, where

$$\hat{r}_i = \begin{cases} r_i & \text{for } i \neq j \\ r_i/2 & \text{for } i = j \end{cases}, \quad c_i^\pm = \begin{cases} c_i & \text{for } i \neq j \\ c_i \pm r_i/2 & \text{for } i = j \end{cases}.$$

Let $P(Q, 0) := Q$, that is, $P(Q, 0) = B(c^0, r^0)$, where

$$c_i^0 = \frac{a_i + b_i}{2}, \quad r_i^0 = \frac{b_i - a_i}{2}, \quad i = 1, \ldots, M.$$

Denote by $\mathcal{P}(Q, d), d \in \mathbb{N}$, the set of boxes obtained after d subdivision steps starting with $B(c^0, r^0)$, where in each step $i = 1, \ldots, d$ the boxes are subdivided with respect to the j_i-th coordinate, where j_i is varied cyclically. That is, $j_i = ((i - 1) \mod n) + 1$. Note that for every point $y \in Q \backslash \partial Q$ and every subdivision step d there exists exactly one box $B = B(y, d) \in \mathcal{P}(Q, d)$ with center c and radius r such that $c_i - r_i \leq y_i < c_i + r_i$, $\forall i = 1, \ldots, M$. Thus, every set of solutions $\mathcal{S}_B$ leads to a set of box collections $\mathcal{B}_d$. These collections can easily be stored in a binary tree with depth d. In Figure 1 a representation of five boxes with subdivision step three and three dimensions ($M = 3$) together with the corresponding set $\mathcal{B}_3$ is shown. Note that each $\mathcal{B}_d$ is completely determined by the tree structure and the initial box $B(c^0, r^0)$. Using this scheme, the memory requirements grow only linearly in the dimension M of the problem.

Multi-Objective Optimization

In a *multi–objective optimization problem* (MOP) the task is to simultaneously optimize k *objective functions* $f_1, \ldots, f_k : \mathbb{R}^n \to \mathbb{R}$. More precisely, a general MOP can be stated as follows:

$$\min_{x \in R} \{F(x)\}, \quad R := \{x \in \mathbb{R}^n \,|\, h(x) = 0, \, g(x) \leq 0\}, \tag{MOP}$$
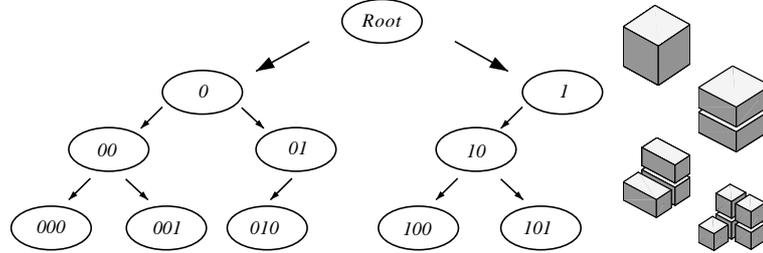
Fig. 1. The data structure used for the representation of the solution set.

where the function F is defined as the vector of the objective functions

$$F : \mathbb{R}^n \to \mathbb{R}^k, \qquad F(x) = (f_1(x), \dots, f_k(x)),$$

and $h : \mathbb{R}^n \to \mathbb{R}^m$, $m \leq n$, and $g : \mathbb{R}^n \to \mathbb{R}^q$. Obviously, we have to define what is meant by finding the minimum of a vector valued function in (MOP). For this we state the following definition.

Definition 1. *(a) Let $v, w \in \mathbb{R}^k$. Then the vector v is* less than w $(v <_p w)$, *if $v_i < w_i$ for all $i \in \{1, \dots, k\}$. The relation $\leq_p$ is defined in an analogous way.*
(b) A vector $v \in \mathbb{R}^k$ is dominated *by a vector $w \in \mathbb{R}^k$ if $w \leq_p v$ and $v \neq w$ (i.e. there exists a $j \in \{1, \dots, k\}$ such that $w_j < v_j$).*

Since $\leq_p$ just defines a partial order on $\mathbb{R}^n$, we cannot proceed as in the classical scalar case. In fact, one cannot expect to find isolated stationary points. Rather one has to find the set of "optimal compromises" and – following Pareto ([34]) – these are defined in the following way.

Definition 2. *(a) Consider the multi–objective optimization problem (MOP). Then a point $\bar{x} \in R$ is called* (globally) Pareto optimal *or a* (global) Pareto point *if there is no $y \in R$ such that*

$$F(y) \neq F(\bar{x}) \quad and \quad F(y) \leq_p F(\bar{x}). \tag{11}$$

(b) A point $\bar{x} \in R$ is a local Pareto point*, if there is a neighborhood $U(\bar{x}) \subset R$ of $\bar{x}$ such that there is no $y \in U(\bar{x})$ satisfying (11).*

Fundamental for most of the methods for the numerical treatment of MOPs is the following theorem of Kuhn and Tucker ([19]) which states a necessary condition for Pareto optimality for MOPs with equality constraints[2].

Theorem 1. *Let x^* be a Pareto point of (MOP) with $q = 0$. Suppose that the set of vectors $\{\nabla h_i(x) \,|\, i = 1, \dots, m\}$ is linearly independent. Then there exist*

---

[2] Without loss of generality we will consider only equality constraints. For a more general formulation of the theorem we refer e.g. to [20].

vectors $\lambda \in \mathbb{R}^m$ *and* $\alpha \in \mathbb{R}^k$ *with* $\alpha_i \geq 0, i = 1, \ldots, k$, *and* $\sum_{i=1}^{k} \alpha_i = 1$ *such that*

$$\sum_{i=1}^{k} \alpha_i \nabla f_i(x^*) + \sum_{j=1}^{m} \lambda_j \nabla h_j(x^*) = 0 \tag{12}$$
$$h_i(x^*) = 0, \ i = 1, \ldots, m.$$

In the unconstrained case – i.e. for $m = 0$ – the theorem claims that the vector of zeros can be written as a convex combination of the gradients of the objectives at every Pareto point. Obviously, (12) is not a sufficient condition for (local) Pareto optimality. On the other hand points satisfying (12) are certainly "Pareto candidates" and thus, following [20], we now emphasize their relevance by the following

Definition 3. *A point* $x \in \mathbb{R}^n$ *is called a* substationary point *or* Karush–Kuhn–Tucker point[3] *(KKT–point) if there exist scalars* $\alpha_1, \ldots, \alpha_k \geq 0$ *and* $\lambda \in \mathbb{R}^m$ *such that (12) is satisfied.*

Having stated Theorem 1, one is in the position to give a qualitative description of the set of Pareto optimal solutions.
Denote by $\tilde{F} : \mathbb{R}^{n+m+k} \to \mathbb{R}^{n+m+1}$ the following auxiliary function:

$$\tilde{F}(x, \lambda, \alpha) = \begin{pmatrix} \sum_{i=1}^{k} \alpha_i \nabla f_i(x) + \sum_{j=1}^{m} \lambda_j \nabla h_j(x^*) \\ h(x) \\ \sum_{i=1}^{k} \alpha_i - 1 \end{pmatrix}. \tag{13}$$

By Theorem 1 it follows that for every substationary point $x^* \in \mathbb{R}^n$ there exist vectors $\lambda^* \in \mathbb{R}^m$ and $\alpha^* \in \mathbb{R}^k$ such that

$$\tilde{F}(x^*, \lambda^*, \alpha^*) = 0. \tag{14}$$

Hence one expects that the set of KKT-points define a $(k-1)$-dimensional manifold due to the Implicit Function Theorem. This is indeed the case under certain smoothness assumptions, see [16] for a thorough discussion of this topic.

3   The Algorithm

In this section we present a continuation algorithm for the computation of the set of KKT-points. Starting with a box collection $\mathcal{B} \subset \mathcal{P}(Q, d)$ of n-dimensional

---

[3] Named after the works of Karush [18] and Kuhn & Tucker [19] for scalar–valued optimization problems.

boxes[4], where every box $B \in \mathcal{B}$ contains at least one computed KKT-point, the aim is to successively extend $\mathcal{B}$ by further boxes which also contain KKT-points. Hence, we can associate with every box $B \in \mathcal{B}$ an approximated solution $a^B \in \mathbb{R}^{n+k+m}$ of (13) , i.e.

$$a^B = (x^B, \alpha^B, \lambda^B), \ \ x^B \in B \quad \text{and} \quad \tilde{F}(x^B, \lambda^B, \alpha^B) \approx 0.$$

Given an initial set $\mathcal{S}_0$ of KKT-points of an MOP and the corresponding box collection $\mathcal{B}$ the algorithm CONT-Recover reads as follows:

Algorithm CONT–Recover

(0) $\mathcal{S}_Q := \mathcal{S}_0$
(1) mark all boxes $B \in \mathcal{B}$.
(2) for all marked boxes $B \in \mathcal{B}$:
    (a) unmark box
    (b) compute a set of orthonormal vectors $\{q_1, \ldots, q_{k-1}\}$ such that
        $span\{q_1, \ldots, q_{k-1}\} = T_{(x^B, \alpha^B, \lambda^B)}\mathcal{M}$.
    (c) generate predictors $s_1, \ldots, s_{n_B} \in T_{(x^B, \alpha^B, \lambda^B)}\mathcal{M}$.
    (d) for $i = 1, \ldots, n_B$:
        starting with s_i, compute $(x^F, \alpha^F, \lambda^F)$ with $\tilde{F}(x^F, \alpha^F, \lambda^F) \approx 0$.
        If $B(x^F, d) \notin \mathcal{B}$: add $B(x^F, d)$ to the collection $\mathcal{B}$, mark the box,
        and set $a^{B(x^F, d)} := (x^F, \alpha^F, \lambda^F)$ and $\mathcal{S}_Q := \mathcal{S}_Q \cup a^{B(x^F, d)}$
Repeat **(2)** while new boxes are added to $\mathcal{B}$ or until a prescribed maximal number of steps is reached.

Remark 1. (a) As described in the previous section, the orthonormal vectors $q_1, \ldots, q_{k-1}$ can be computed via a QR-decomposition of $\tilde{F}'(x^B, \alpha^B, \lambda^B)^T$. If $\dim T_{(x^B, \alpha^B, \lambda^B)}\mathcal{M} < k-1$, then $\mathcal{M}$ is not a $(k-1)$-dimensional manifold in the neighborhood of the KKT-point $(x^B, \alpha^B, \lambda^B)$. In this case we continue the search in all coordinate directions, i.e. we take $q_i = e_i$, $i = 1, \ldots, m$, where e_i is the ith unit vector of $\mathbb{R}^m$.
(b) A note on the predictor step: for an approximation $a^B = (x^B, \alpha^B, \lambda^B)$, which is associated with a box B, we select predictors $s = (x^s, \alpha^s, \lambda^s)$ where x^s is contained in a neighboring box of B, i.e. in a box $\hat{B}$ with $B \cap \hat{B} \neq \emptyset$.
(c) Since no particular structure of $\tilde{F}$ is taken into account the algorithm can be used to compute general implicitly defined sets $H^{-1}(0)$ of functions

$$H : \mathbb{R}^{N+K} \to \mathbb{R}^N.$$

In the situation where H is only continuous the algorithm stated above can still be applied successfully using the following modifications:

---

[4] Since the corresponding vectors α and λ of a KKT-point x (see Def. 3) are only needed to generate further KKT-points and are not required any further, we can restrict ourselves to a domain $Q = [a_1, b_1] \times \ldots, [a_n, b_n] \subset \mathbb{R}^n$ of the parameter space of the given MOP.

- The predictors can be chosen e.g. as $q_i = e_i, i = 1, \ldots, n$, as described in part (a) of this remark.
- Tor the corrector step, in principle any derivative free minimization algorithm applied on $\|H(x)\|$ can be used, e.g. the *downhill simplex method* of Nelder and Mead, see [22] or [23].

4   Numerical Results

In this section we illustrate the efficiency of the algorithm by several examples.

4.1   Example 1

First we consider the following unconstrained MOP:

$$f_1, f_2, f_3 : \mathbb{R}^n \to \mathbb{R}$$
$$f_i(x) = \sum_{\substack{j=1 \\ j \neq i}}^{n} (x_j - a_j^i)^2 + (x_i - a_i^i)^4, \tag{15}$$

where

$$
\begin{aligned}
a^1 &= (1, 1, 1, 1, \ldots) & \in \mathbb{R}^n \\
a^2 &= (-1, -1, -1, -1, \ldots) & \in \mathbb{R}^n \\
a^3 &= (1, -1, 1, -1, \ldots) & \in \mathbb{R}^n
\end{aligned}
$$

Since all objectives are strictly convex, the set of KKT-points is equal to the Pareto set and consists of one connected component (see e.g. [20]). Figure 2 shows the result for $n = 100$ and $n = 1000$ (in image space). In these cases the algorithm CONT-Recover was started with two given KKT-points.

4.2   Example 2

Next we consider a multi–objective optimization problem consisting of four objectives. In fact, it is an augmented model of MOP (15):
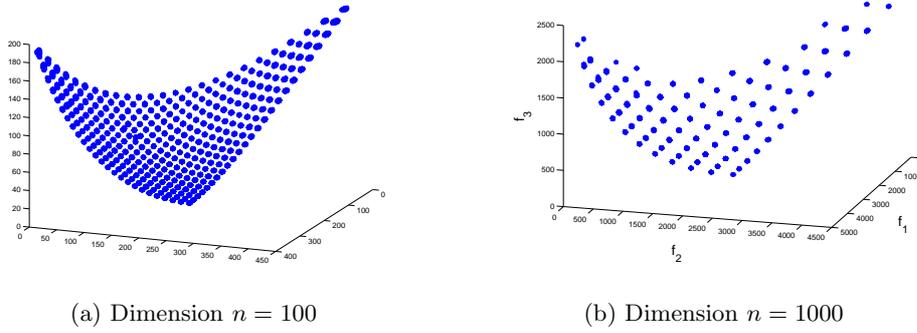
$$f_1, f_2, f_3, f_4 : \mathbb{R}^n \to \mathbb{R}$$
$$f_i(x) = \sum_{\substack{j=1 \\ j \neq i}}^{n} (x_j - a_j^i)^2 + (x_i - a_i^i)^4, \tag{16}$$

where

$$
\begin{aligned}
a^1 &= (\ \ 1, \ \ 1, \ \ 1, \ \ 1, \ldots) \in \mathbb{R}^n \\
a^2 &= (-1, -1, -1, -1, \ldots) \ \ \in \mathbb{R}^n \\
a^3 &= (\ \ 1, -1, \ \ 1, -1, \ldots) \ \ \in \mathbb{R}^n \\
a^4 &= (\ \ 1, \ \ 1, -1, -1, \ldots) \ \ \in \mathbb{R}^n
\end{aligned}
$$

Figures 3 and 4 show computational results for dimension $n = 10$ in parameter space and in image space.

(a) Dimension $n = 100$ (b) Dimension $n = 1000$

Fig. 2. Pareto sets of MOP (15) in image space.

4.3 Example 3

In this example we make an insertion from the field of multi–objective optimization and turn our attention to the computation of sets $H^{-1}(0)$ of continuous functions $H : \mathbb{R}^n \to \mathbb{R}^m$. In the following we consider

$$H_1, H_2 : \mathbb{R}^3 \to \mathbb{R}^1$$

$$H_1(x, y, z) := x^4 - 3xy - \cos(4z) + \cos(xy) \tag{17}$$

$$H_2(x, y, z) := \min\{|x|, |y|, |z|\}$$

The coverings of the 2-manifolds $H_1^{-1}(0)$ and $H_2^{-1}(0)$ are shown in Figure 5. Since H_1 is continuously differentiable we have used the Gauss-Newton method for the corrector step, while for the latter function we have used the downhill simplex method (see Rmk. 1).
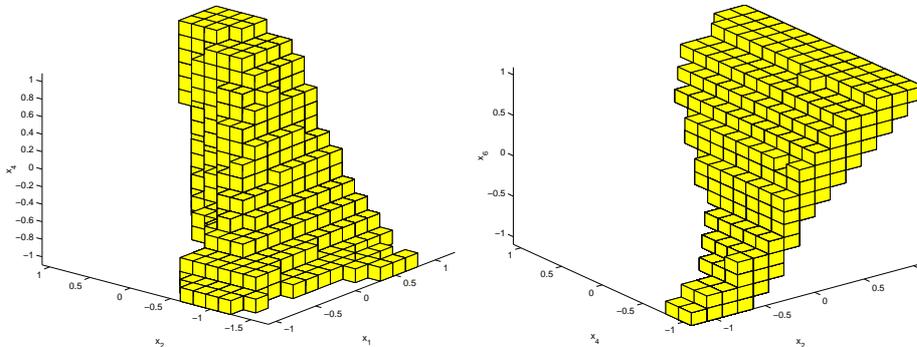
4.4 Example 4

Finally, we turn our attention to the following constrained MOP:

$$\min F(x) := \begin{pmatrix} (x_1 - 1)^4 + (x_2 - 1)^2 + (x_3 - 1)^2 \\ (x_1 + 1)^2 + (x_2 + 1)^4 + (x_3 + 1)^2 \\ (x_1 - 1)^2 + (x_2 + 1)^2 + (x_3 - 1)^4 \end{pmatrix} \tag{18}$$

subject to the equality constraint

$$h(x) = r^2 - z^2 - (\sqrt{x^2 + y^2} - R)^2 = 0.$$

Fig. 3. Computation of MOP (16): The figures show two projections of the covering of the Pareto set in parameter space for $n = 10$.

Figure 6 shows both the set of KKT-points and the Pareto set for the following values:

$$Q = [-1, 1]^3, \quad z = 0, \quad r = 0.3, \quad R = 0.5.$$

We have obtained the set of global Pareto points by a combination of the continuation method described above with the following archiving strategy (see also [30]): In the initial step, an archive $\mathcal{A}$ is generated consisting of all nondominated points of $\mathcal{S}_0$. In the course of the computation this archive gets permanently updated by the "candidates" x^F which were produced by the continuation method. In addition, only boxes are added to the collection $\mathcal{B}$ where its representative is nondominated in $\mathcal{A}$ (i.e., according to the set of previously considered candidates). For the storage and update of the archive we have used the data structure presented in [29].

5   Conclusion

We have demonstrated the applicabilty of continuation methods for the numerical treatment of MOPs. Further, we have presented a new predictor-corrector variant using boxes for the representation of the set of KKT-points. This data structure allows an effective representation of the entire set of interest, in particular in higher dimensions. Finally, we have presented some numerical results indicating the strength of the approach.

References

1. E. L. Allgower and K. Georg. *Numerical Continuation Methods.* Springer, 1990.
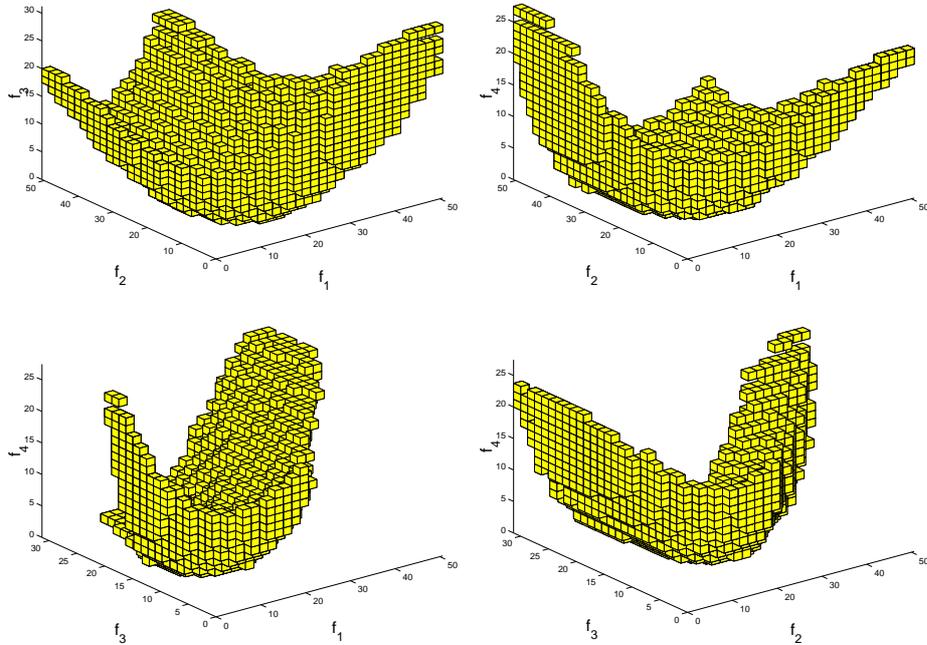
Fig. 4. Computation of MOP (16): The figures show all projections of the Pareto set in the image space for $n = 10$.

2. E. L. Allgower and P. H. Schmidt. An algorithm for piecewise-linear approximation of an implicitly defined manifold. *SIAM Journal of Numerical Analysis*, 22:322–346, 1987.

3. C. A. Coello Coello and M. S. Lechunga. MOPSO: A proposal for multiple objective particle swarm optimization. In *Proceedings of the IEEE World Congress on Computational Intelligence*, IEEE Press, 2002.

4. I. Das and J. Dennis. *Normal-boundary intersection*: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal of Optimization*, 8:631 – 657, 1998.

5. K. Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. Wiley, 2001.

6. K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *Parallel Problem Solving from Nature VI (PPSN-VI)*, pages 849–858, 2000.

7. K. Deb, S. Pratap, and A. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pages 849–858. Springer, 2000.

8. M. Dellnitz, O. Schütze, and T. Hestermeyer. Covering pareto sets by multilevel subdivision techniques. *Journal of Optimization Theory and Applications*, 124:113–155, 2005.

(a) $H_1^{-1}(0)$ (b) $H_2^{-1}(0)$
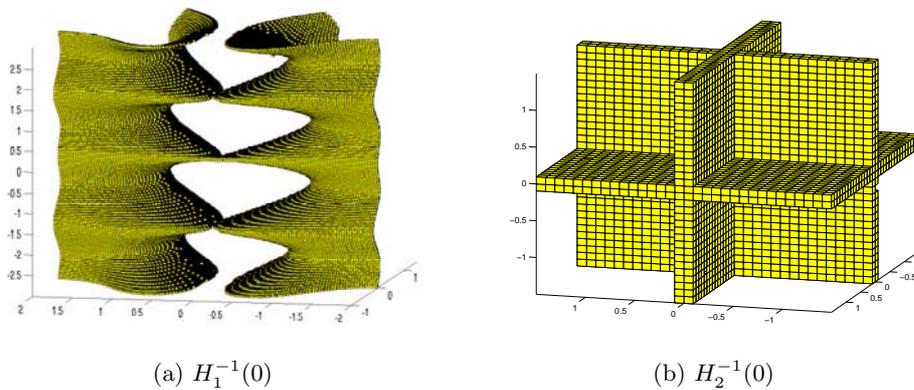
Fig. 5. Computation of general implicitly defined manifolds (see (17)).

 9. M. Dellnitz, O. Schütze, and St. Sertl. Finding zeros by multilevel subdivision techniques. *IMA Journal of Numerical Analysis*, 22(2):167–185, 2002.
10. P. Deuflhard. *Newton Methods for Nonlinear Problems. Affine Invariance and Adaptive Algorithms*. Springer, 2004.
11. J. E. Fieldsend and S. Singh. A multi-objective algorithm based upon particle swarm optimization, an efficient data structure and turbulence. In *Proceedings of the 2002 U.K. Workshop on Computational Intelligence*, 2002.
12. J. Fliege and A. Heseler. Constructing approximations to the efficient set of convex quadratic multiobjective problems. University of Dortmund, Germany, Technical Report, 2003.
13. C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele, editors. *Evolutionary Multi-Criterion Optimization. Second International Conference, EMO 2003*. Springer, 2003.
14. J. Guddat, F. Guerra Vasquez, K. Tammer, and K. Wendler. *Multiobjective and Stochastic Optimization based on Parametric Optimization*. Akademie-Verlag, 1985.
15. M. E. Henderson. Multiple parameter continuation: Computing implicitly defined k-manifolds. *International Journal of Bifurcation and Chaos*, 12:451–476, 2002.
16. C. Hillermeier. *Nonlinear Multiobjective Optimization - A Generalized Homotopy Approach*. Birkhäuser, 2001.
17. J. Jahn. *Mathematical Vector Optimization in Partially Ordered Linear Spaces*. Verlag Peter Lang GmbH, Frankfurt am Main, 1986.
18. W. E. Karush. *Minima of functions of several variables with inequalities as side conditions*. PhD thesis, University of Chicago, 1939.
19. H. Kuhn and A. Tucker. Nonlinear programming. In J. Neumann, editor, *Proceeding of the 2nd Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, 1951.
20. K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.
21. S. Mostaghim. *Multi-Objective Evolutionary Algorithms, Data Structures, Convergence and Diversity*. PhD thesis, University of Paderborn, 2004.
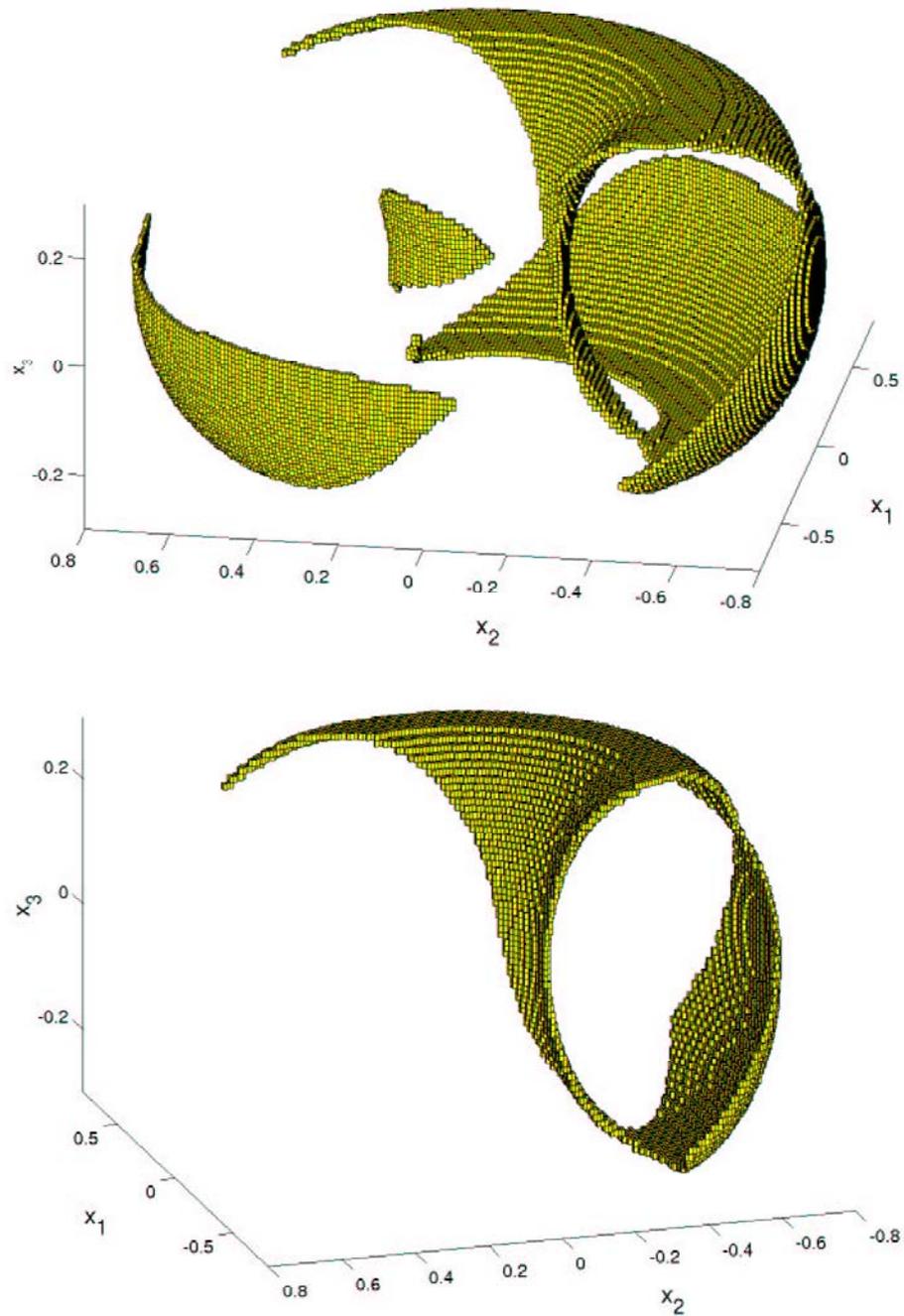
Fig. 6. Computation of MOP (18): set of substationary points (above) and Pareto points (below).

22. J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.
23. R. P. Brent. *Algorithms for Minimization without Derivatives*. Englewood Cliffs, NJ, 1973.
24. J. Rakowska, R. T. Haftka, and L. T. Watson. Tracing the efficient curve for multi-objective control-structure optimization. *Computing Systems in Engineering*, 2(6):461–471, 199.
25. W. Rheinboldt. On the computation of multi-dimensional solution manifolds of parametrized equations. *Numer. Math.*, 53:165–181, 1988.
26. W. C. Rheinboldt. *Numerical Analysis of Parametrized Nonlinear Equations*. Wiley, 1986.
27. S. Schäffler, R. Schultz, and K. Weinzierl. A stochastic method for the solution of unconstrained vector optimization problems. *Journal of Optimization Theory and Applications*, 114(1):209–222, 2002.
28. K. Schittkowski. EASY-OPT: An interactive optimization system with automatic differentiation - user's guide. Department of Mathematics, University of Bayreuth, Technical Report, 1999.
29. O. Schütze. A new data structure for the nondominance problem in multi-objective optimization. In C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele, editors, *Evolutionary Multi-Criterion Optimization (EMO 03)*, volume 2 of *Springer*, 2003.
30. O. Schütze. *Set Oriented Methods for Global Optimization*. PhD thesis, University of Paderborn, 2004.
31. O. Schütze, S. Mostaghim, M. Dellnitz, and J. Teich. Covering Pareto sets by multilevel evolutionary subdivision techniques. In C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele, editors, *Evolutionary Multi-Criterion Optimization*, Lecture Notes in Computer Science, 2003.
32. R. E. Steuer. *Multiple Criteria Optimization: Theory, Computation, and Applications*. John Wiley & Sons, Inc., 1986.
33. R. E. Steuer, editor. *Multiple Criteria Decision Making and Risk Analysis Using Microcomputers*, chapter The Tchebycheff Procedure of Interactive Multiple Objective Programming, pages 235–249. Springer, Berlin, 1989.
34. V. Pareto. *Cours d'Economie Politique*. Libraire Droz, Genève, 1964 (first edition in 1896).
35. E. Zitzler, K. Deb, L. Thiele, C. A. Coello Coello, and D. Corne, editors. *Evolutionary Multi-Criterion Optimization. First International Conference, EMO 2001*. Springer, 2001.
36. E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm. In *Evolutionary Methods for Design, Optimisation and Control with Applications to Industrial Problems*, pages 95–100. CIMNE, Barcelona, Spain, 2002.